An asymmetric primitive based on the Bivariate Function Hard Problem

Muhammad Rezal Kamel Ariffin[1,2]

[1] Al-Kindi Cryptography Research Laboratory, Institute for Mathematical Research, Universiti Putra Malaysia (UPM)
[2] Department of Mathematics, Faculty of Science, Universiti Putra Malaysia (UPM), Selangor, Malaysia

rezal@putra.upm.edu.my

Abstract. The Bivariate Function Hard Problem (BFHP) has been in existence implicitly in almost all number theoretic based cryptosystems. This work defines the BFHP in a more general setting and produces an efficient asymmetric cryptosystem. The cryptosystem has a complexity order of $O(n^2)$ for both encryption and decryption.

1 Introduction

In Section 2 of this work, we define the Bivariate Function Hard Problem (BFHP) and illustrate its existence within the RSA hard problem. We then proceed to produce an asymmetric cryptosystem in Section 3 that thoroughly utilizes the BFHP concept. In Section 4 we produce a table of comparison between known asymmetric algorithms and the algorithm introduced in this work. We conclude in Section 5.

2 Bivariate Function Hard Problem

The following proposition gives a proper an analytical description of the "Bivariate Function Hard Problem" (BFHP).

Proposition 1. *Let $F(x_1, x_2, ..., x_n)$ be a multivariate one-way function that maps $F : \mathbb{Z}^n \to \mathbb{Z}^+_{(2^{n-1},2^n-1)}$. Let F_1 and F_2 be such functions (either identical or non-identical) such that $A_1 = F_1(x_1, x_2, ..., x_n)$, $A_2 = F_2(y_1, y_2, ..., y_n)$ and $gcd(A_1, A_2) = 1$. Let $u, v \in \mathbb{Z}^+_{(2^{m-1},2^m-1)}$. Let*

$$G(u, v) = A_1 u + A_2 v \tag{1}$$

If at minimum $m - n - 1 = 129$, it is infeasible to determine (u, v) from $G(u, v)$. Furthermore, (u, v) is unique for $G(u, v)$ with high probability.

Remark 1. Before we proceed with the proof, we remark here that the diophantine equation given by $G(u, v)$ is solved when the parameters (u, v) are found. That is, the BFHP is solved when the parameters (u, v) are found.

Proof. We begin by proving that (u, v) is unique for each $G(u, v)$ with high probability. Assume there exists $u_1 \neq u_2$ and $v_1 \neq v_2$ such that

$$A_1 u_1 + A_2 v_1 = A_1 u_2 + A_2 v_2 \tag{2}$$

We will then have

$$Y = v_1 - v_2 = \frac{A_1(u_1 - u_2)}{A_2}$$

Since $\gcd(A_1, A_2) = 1$ and $A_2 \approx 2^n$, then the probability that Y is an integer is 2^{-n}.

Next we proceed to prove that to solve the diophantine equation given by $G(u, v)$ is infeasible to be solved. The general solution for $G(u, v)$ is given by

$$u = u_0 + A_2 t \tag{3}$$

and

$$v = v_0 - A_1 t \tag{4}$$

for some integer t. To find u within the stipulated interval $(u \in (2^{m-1}, 2^m - 1))$ we have to find the integer t such that $2^{m-1} < u < 2^m - 1$. This gives

$$\frac{2^{m-1} - u_0}{A_2} < t < \frac{2^m - 1 - u_0}{A_2}.$$

Then the difference between the upper and the lower bound is

$$\frac{2^m - 1 - 2^{m-1}}{A_2} = \frac{2^{m-1} - 1}{A_2} \approx \frac{2^{m-2}}{2^n} = 2^{m-n-2}.$$

Since $m - n - 1 = 129$, then $m - n - 2 = 128$. Hence the difference is very large and finding the correct t is infeasible. This is also the same scenario for v.$\blacksquare$

Remark 2. It has to be noted that the BFHP in the form we have described has to be coupled with other mathematical considerations upon F_1, F_2, u, v to yield practical cryptographic constructions.

Definition 1. *Let the tuple (M, e, d, p, q) be strong RSA parameters. Let $N = pq, ed \equiv 1(mod\ \phi(N))$ and $\phi(N) = (p-1)(q-1)$. From $C \equiv M^e(mod\ N)$ we rewrite as*

$$C(M, j) = M^e - Nj \tag{5}$$

where j is the number of times M^e is reduced by N until $C(M, j)$ is obtained. The problem of determining (M, j) from equation (5) is the RSA BFHP. The pair (M, j) is unique with high probability for each $C(M, j)$.

Remark 3. With little effort, one can also produce a BFHP for the discrete log problem (DLP). Analysis could also be done within the framework given for the RSA-BFHP.

The following 3 analytical results gives a clear picture regarding the RSA-BFHP. All result re-affirms the "infeasibility" of trying solve the RSA problem. We also produce a corollary that may shed some light regarding the RSA problem and integer factorization.

Lemma 1. *The RSA BFHP is infeasible to be solved.*

Proof. Let $X = M^e$. From

$$C(X, j) = X - Nj \tag{6}$$

the general solution is

$$X = X_0 - Nt$$

and

$$j = j_0 + t$$

for some $t \in \mathbb{Z}$. It is easy to deduce that the correct t belongs in the interval $(2^{k(e-1)-1}, 2^{k(e-1)} - 1)$. Current RSA deployment has $k = 1024$. Hence, to solve the RSA BFHP is infeasible.■

Lemma 2. *RSA problem* $\equiv_p$ *RSA BFHP*

Proof. From $C \equiv M^e (\bmod N)$ if the RSA problem is solved then M is found. Hence, $j = \frac{M^e - C}{N}$ is also found. Thus, the RSA BFHP is solved.

From $C(X, j) = X - Nj$, if the RSA BFHP is solved means that (M, j) is found. Thus, the RSA problem is solved.■

Corollary 1. *Solving RSA BFHP does not imply successful factoring of* $N = pq$.

Proof. From Remark 1, if RSA BFHP is solved then (M, j) is found. That is,

$$M = \sqrt[e]{C + Nj}$$

and

$$j = \frac{M^e - C}{N}.$$

It is obvious that the factoring of N was not obtained.■

3 A new asymmetric cryptosystem based on the BFHP

3.1 Common values

This scheme is to facilitate secure communication asymmetrically between 2 parties namely A (Along) and B (Busu). For both of them there will 2 sets of public parameters determined pre-communication and a common n-bit prime number. The party that initiates the communication will utilize the set $G_1 = (g_1, g_2)$

while the other party will utilize the set $G_2 = (g_3, g_4)$. These public parameters are co-prime to each other and belong in the interval $(2^{n-1}, 2^n - 1)$. In fact both parties will have keys generated by both sets for the eventuality of either initiating communication or accepting incoming information. In this work we assume Along is initiating while Busu is accepting secure information.

- **Key Generation by Along -sender**

INPUT: The public prime number p, the public sets G_1 and G_2.
OUTPUT: A public key for sending information e_A, an ephemeral private key d_A for generating e_A and a secret pair (α_1, α_2).

1. Generate a random private key d_A within the interval $(2^{n-1}, 2^n - 1)$.
2. Compute $\tilde{\alpha_1} \equiv g_1 d_A (\text{mod } p)$.
3. Compute $\tilde{\alpha_2} \equiv g_2 d_A (\text{mod } p)$.
4. Generate two random and distinct n-bit ephemeral keys k_{A1} and k_{A2}.
5. Compute the secret integers $\alpha_1 = \tilde{\alpha_1} + k_{A1}p$ and $\alpha_2 = \tilde{\alpha_2} + k_{A2}p$. Both α_1 and α_2 belong in the interval $(2^{2n-1}, 2^{2n} - 1)$.
6. Let $e_A = g_3\alpha_1 + g_4\alpha_2$.

- **Key Generation by Busu -recipient**

INPUT: The public prime number p, the public sets G_1 and G_2.
OUTPUT: A public key for receiving information e_B, an ephemeral private key d_B for generating e_B and a secret pair (β_1, β_2).

1. Generate a random private key d_B within the interval $(2^{n-1}, 2^n - 1)$.
2. Compute $\tilde{\beta_1} \equiv g_3 d_B (\text{mod } p)$.
3. Compute $\tilde{\beta_2} \equiv g_4 d_B (\text{mod } p)$.
4. Generate two random and distinct n-bit ephemeral keys k_{B1} and k_{B2}.
5. Compute the secret integers $\beta_1 = \tilde{\beta_1} + k_{B1}p$ and $\beta_2 = \tilde{\beta_2} + k_{B2}p$. Both β_1 and β_2 belong in the interval $(2^{2n-1}, 2^{2n} - 1)$.
6. Let $e_B = g_1\beta_1 + g_2\beta_2$.

- **Encryption by Along**

INPUT: The public key tuple (e_A, e_B), and the message $\mathbf{M}$ which is n-bits long and less than p.

OUTPUT: The ciphertext C.

1. Upon informing Busu of the intention to send secure data, Along receives Busu's public key e_B.
2. Along then generates $e_{AB} \equiv d_A e_B \equiv (\text{mod } p)$.
3. Along then generates the ciphertext $C_1 = (M + e_{AB})(\text{mod } p)$.

4. Next, Along produces $sk = H(e_{AB})$ where H is a collision resistant hash function.
5. Along will then utilize a symmetric algorithm Enc, to produce $C_2 = Enc_{sk}(M)$.
6. Along will relay (C_1, C_2, e_A) to Busu.

- **Decryption by Busu**

INPUT: The private key d_B and the tuple (C_1, C_2, e_A).

OUTPUT: The message **M**.

1. Upon receiving ciphertext Busu computes $e_{BA} \equiv d_B e_A (\text{mod } p)$.
2. Busu then computes **M'**$=(C_1 - e_{BA})(\text{mod } p)$.
3. Busu then produces $sk = H(e_{BA})$
4. Busu then decrypts C_2 with its corresponding symmetric decryption algorithm Dec to produce $M = Dec_{sk}(C_2)$.
5. If **M'**$\neq M$ then abort.
6. Else output **M'** which is the message.

Proposition 2. *From the above mentioned algorithm* $e_{AB} = e_{BA}$.

Proof. $e_{AB} \equiv d_A e_B \equiv (\tilde{\alpha_1}\tilde{\beta_1} + \tilde{\alpha_2}\tilde{\beta_2}) \equiv (\tilde{\beta_1}\tilde{\alpha_1} + \tilde{\beta_2}\tilde{\alpha_2}) \equiv d_B e_A (\text{mod } p) = e_{BA}$.■

Proposition 3. *The encryption process as mentioned above is IND-CCA2 secure.*

Proof. This is a sketch. Any change to C_1 would result in the decrypted value from C_1 which is M' which would differ from $M = Dec_{sk}(C_2)$ with high probability. Hence, abort.■

Lemma 3. *The problem of determining the secret parameters either from* e_A *or* e_B *is a BFHP.*

Proof. It is obvious that with high probability ($\approx 2^{-n}$ - since each g_i are coprime to each other) that the secret parameters in either e_A or e_B are unique. Also, the difference between the secret and public parameters are n-bits, one can set $n = 128$.■

4 Table of Comparison

Let $|E|$ denote public key size. Let $|M|$ denote the message size. For RSA and ECC we utilize its parameters within its IND-CCA2 design. In determining the ciphertext size $|C|$ we also included the public keys to be transmitted (where applicable). Complexity time is taken in base case scenario deployed via the Fast Fourier Transform (FFT).

Algorithm	Encryption Speed	Decryption Speed	Ratio $\|M\| : \|C\|$	Ratio $\|M\| : \|E\|$	Remark
RSA	$O(n^2\ log\ n)$	$O(n^2\ log\ n)$	$1:2$	$1:2$	2 parameter ciphertext of n-bits each
ECC	$O(n^2\ log\ n)$	$O(n^2\ log\ n)$	$1:3$	$1:2$	2 parameter ciphertext of n-bits each + 1 n-bit public key
NTRU	$O(n\ log\ n)$	$O(n\ log\ n)$	Varies [2]	N/A	
This work	$O(n\ log\ n)$	$O(n\ log\ n)$	$1:5$	$1:3$	2 parameter ciphertext of n-bits each + 1 $3n$-bit public key

Table 1. Comparison table for input block of length n

5 Conclusion

We conclude this work by stating that an efficient asymmetric algorithm has been disclosed. By having complexity order of $O(n\ log\ n)$ for both encryption and decryption, it would cut $\approx \frac{2}{3}$ of the running time of algorithms that do not achieve this speed. Furthermore, it achieves IND-CCA2 security.

References

1. J. Hoffstein, J. Pipher, J. H. Silverman. "NTRU: A Ring Based Public Key Cryptosystem in Algorithmic Number Theory," *Lecture Notes in Computer Science 1423*, pp. 267–288, 1998.
2. J. Hoffstein, D.Lieman, J. Pipher and J. H. Silverman, *NTRU : A Public Key Cryptosystem, NTRU Cryptosystems Inc.*[Online]. Available: http://grouper.ieee.org/groups/1363/lattPK/submissions/ntru.pdf 2008.